Symmetry aware generation of two-staged particle decays in high-energy physics

Niclas Eich, Martin Erdmann, Benjamin Fischer, Niklas Zimmermann
RWTH Aachen University (DE) niclas.eich@rwth-aachen.de

ACAT2021

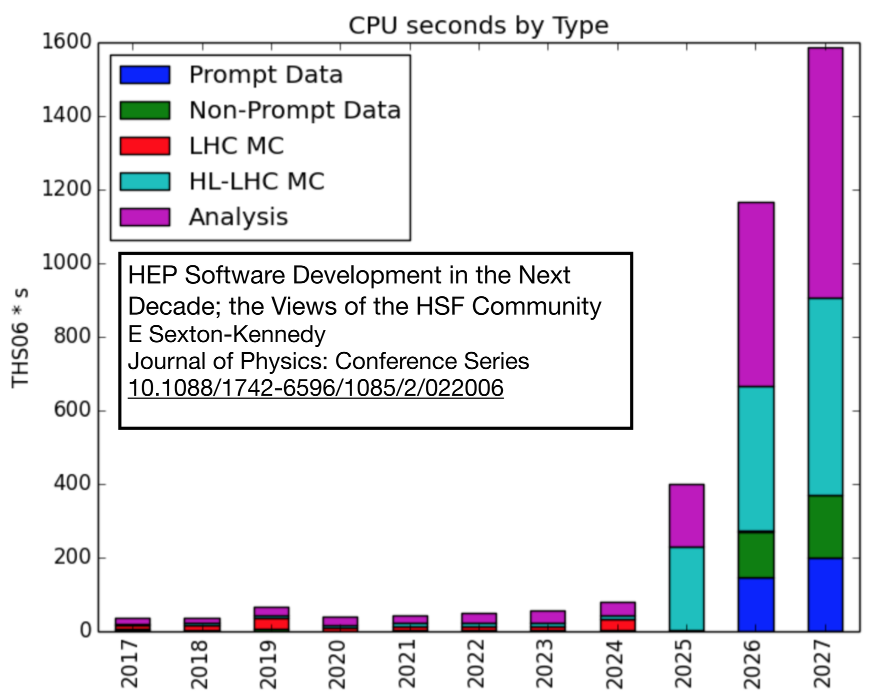
RWTH AACHEN UNIVERSITY
DFG Deutsche Forschungsgemeinschaft
Bundesministerium für Bildung und Forschung

The Challenge

- Large amount of HL-LHC computing resources will be required for MC simulation

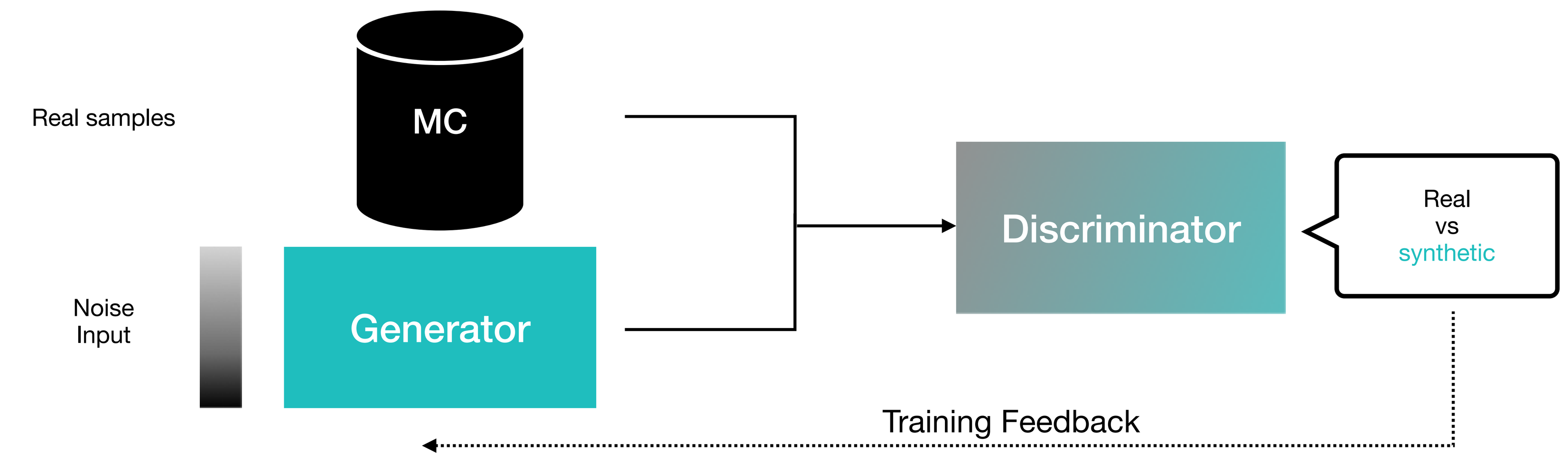
➜ Need for fast simulation methods

Scope:
4-vectors of collision event final states targeted towards analysis



Generative Adversarial Networks

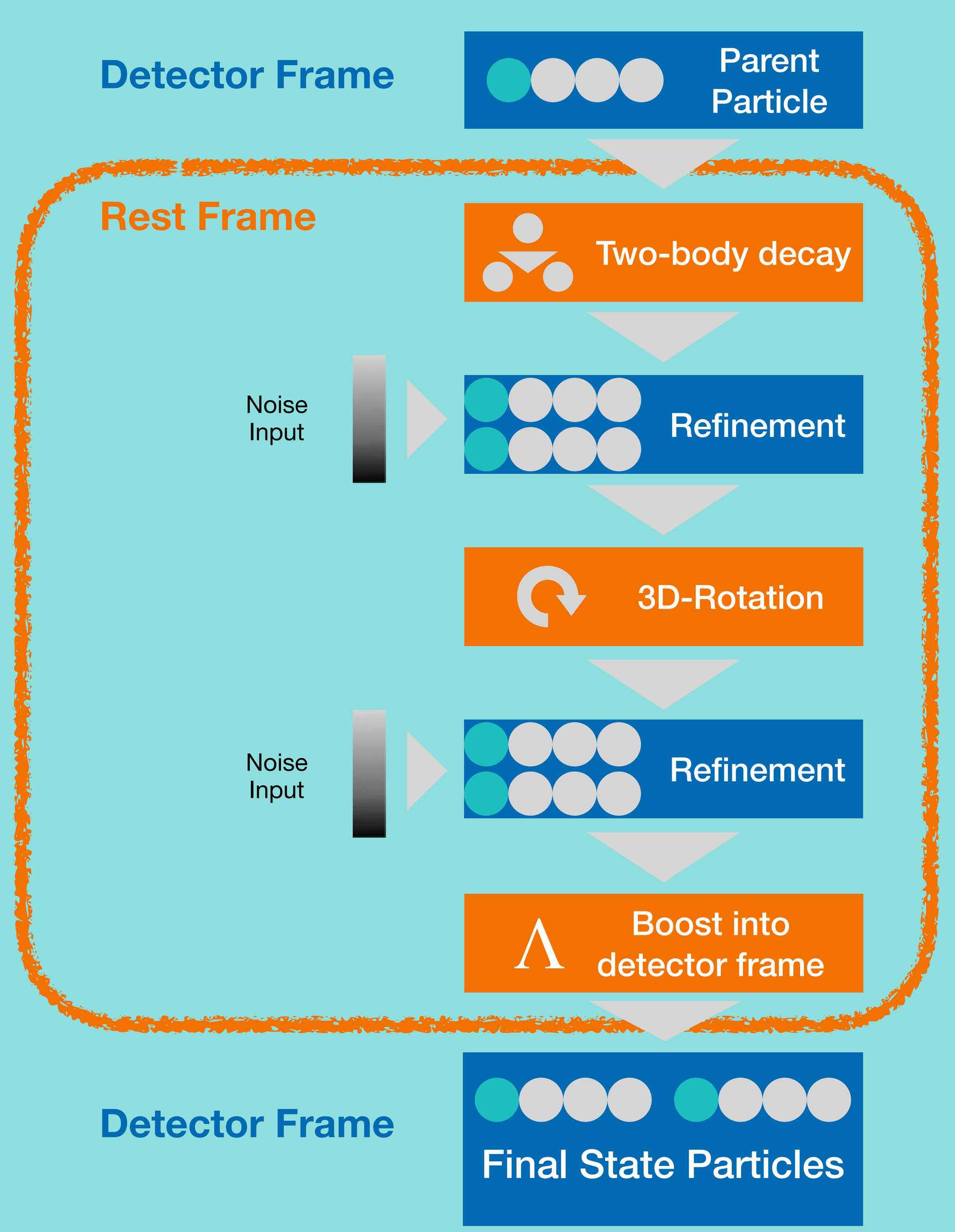
Unsupervised learning method for sample synthesis



Network uses Physicist's Knowledge

Particle Decay Unit (PDU)

Specialised layer utilises analytical physics transformations together with standard machine learning blocks

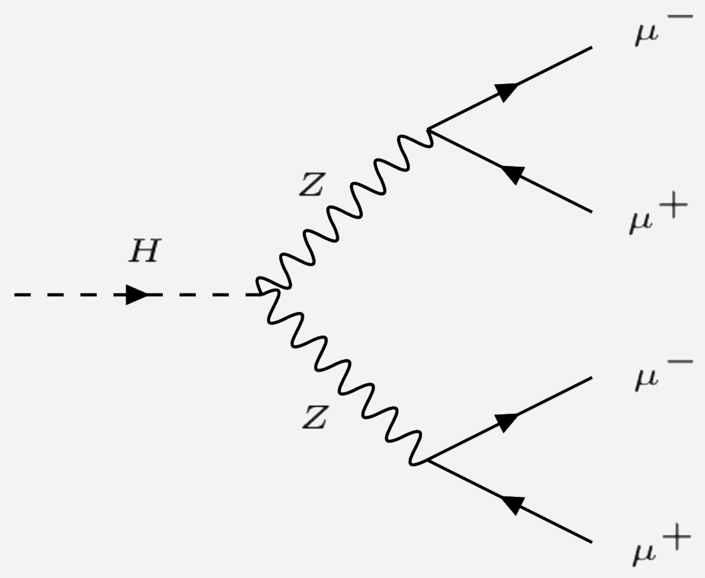


Physics transformations to model inherent symmetries
- Energy/Momentum-conservation
- Rotational symmetries
- Lorentz symmetry

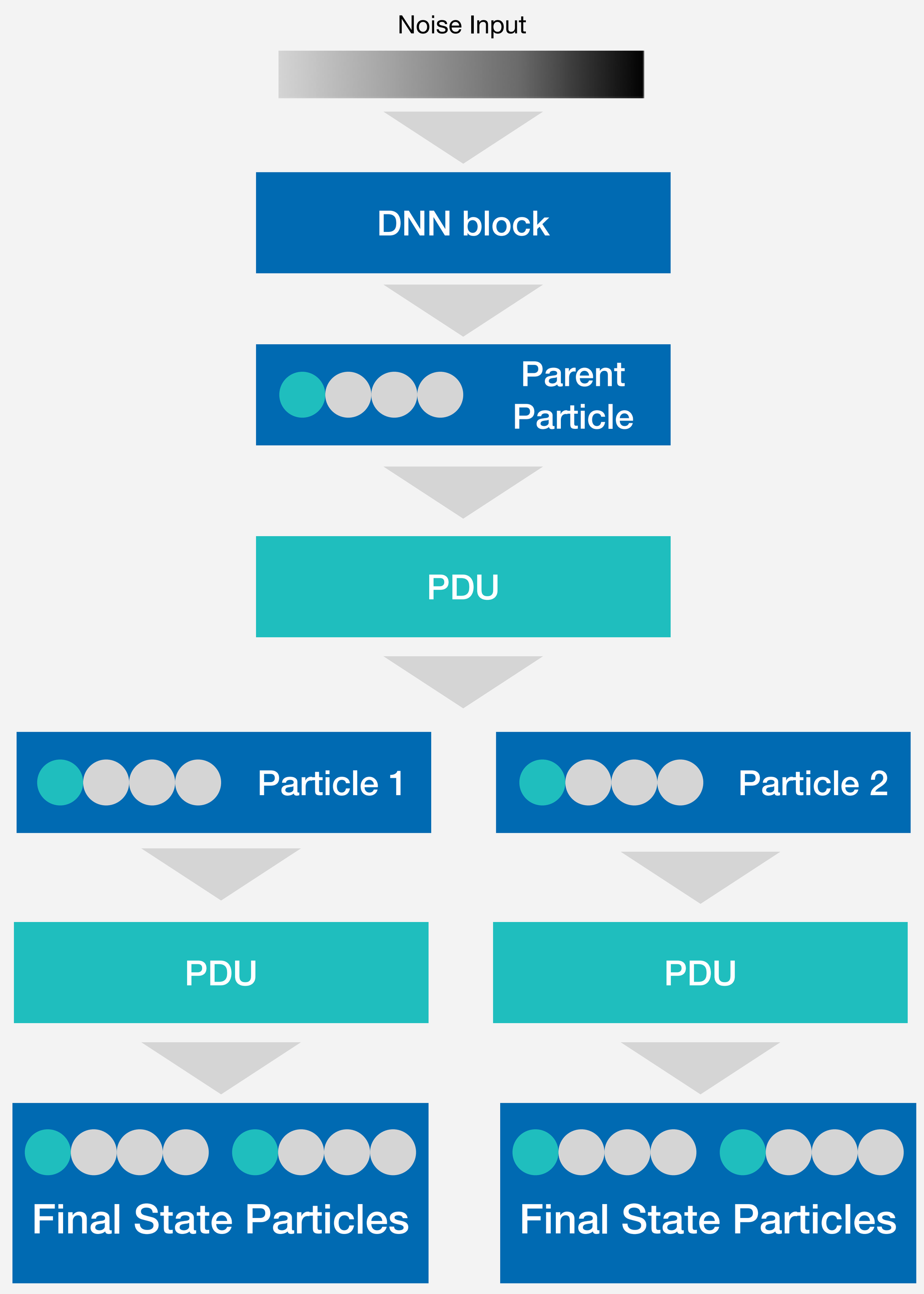
DNN blocks for 4-vector refinement e.g.
- detector effects
- final state radiation

Two-stage decay modelling

Stacking of PDU for modelling decay chains

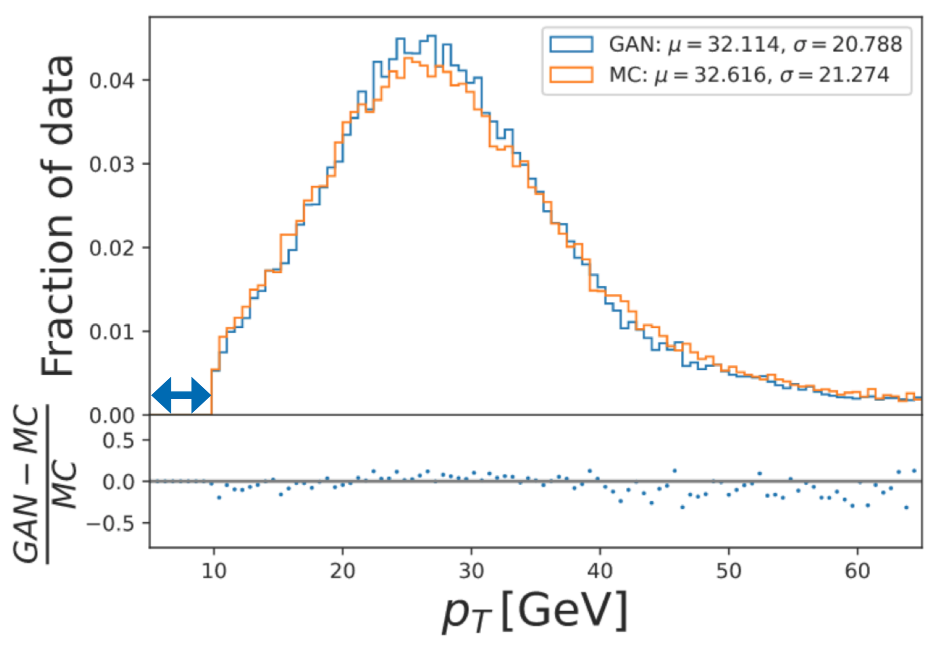


- Stacking of PDUs can preserve symmetries/correlations along the decay chain
- PDU utilised as network-layer
  - Input: one 4-vector
  - Output: two 4-vectors
- DNN-components (e.g. fully connected layers) have the power to model different effects throughout the network on 4-vectors beyond analytic modelling



Training

- Discriminator network inputs
  - 4-vectors of final state particles
  - 4-vectors of reconstructed resonances
  - Reconstructed observables (η, p_T, ϕ, m)
- Generator can produce final states with arbitrary p_T/η values due to rotation
  ➜ Turn off gradients for events outside of cut-regions
  - $p_T > 10GeV$
  - $|\eta| < 2.4$



- Sharp features like mass-peaks are known to be hard to model
  ➜ "How to GAN LHC events" by *Plehn et. al* [10.21468/scipostphys.7.6.075]
  - Modified GAN-Loss

$$L_{\text{Regularised}} = L - \frac{\gamma}{2}\Omega_{JS}(P_r, P_\theta; D)$$

$$\Omega_{JS} = \mathbf{E}_{x\sim P_r}\left[(1-D(x))^2 \parallel \nabla\phi(x) \parallel^2\right] + \mathbf{E}_{x\sim P_\theta}\left[(D(x))^2 \parallel \nabla\phi(x) \parallel^2\right]$$

  - Additional MMD-Kernel-Loss to enforce mass-peak

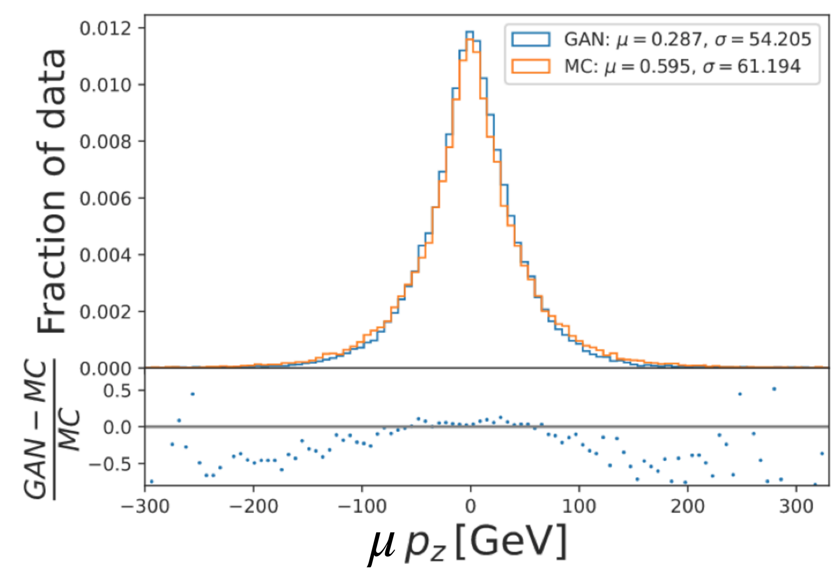
$$\text{MMD}^2(P_T, P_G) = \langle k(x,x')\rangle_{x,x'\sim P_T} + \langle k(y,y')\rangle_{y,y'\sim P_G} - 2\langle k(x,y)\rangle_{x\sim P_T, y\sim P_G}$$

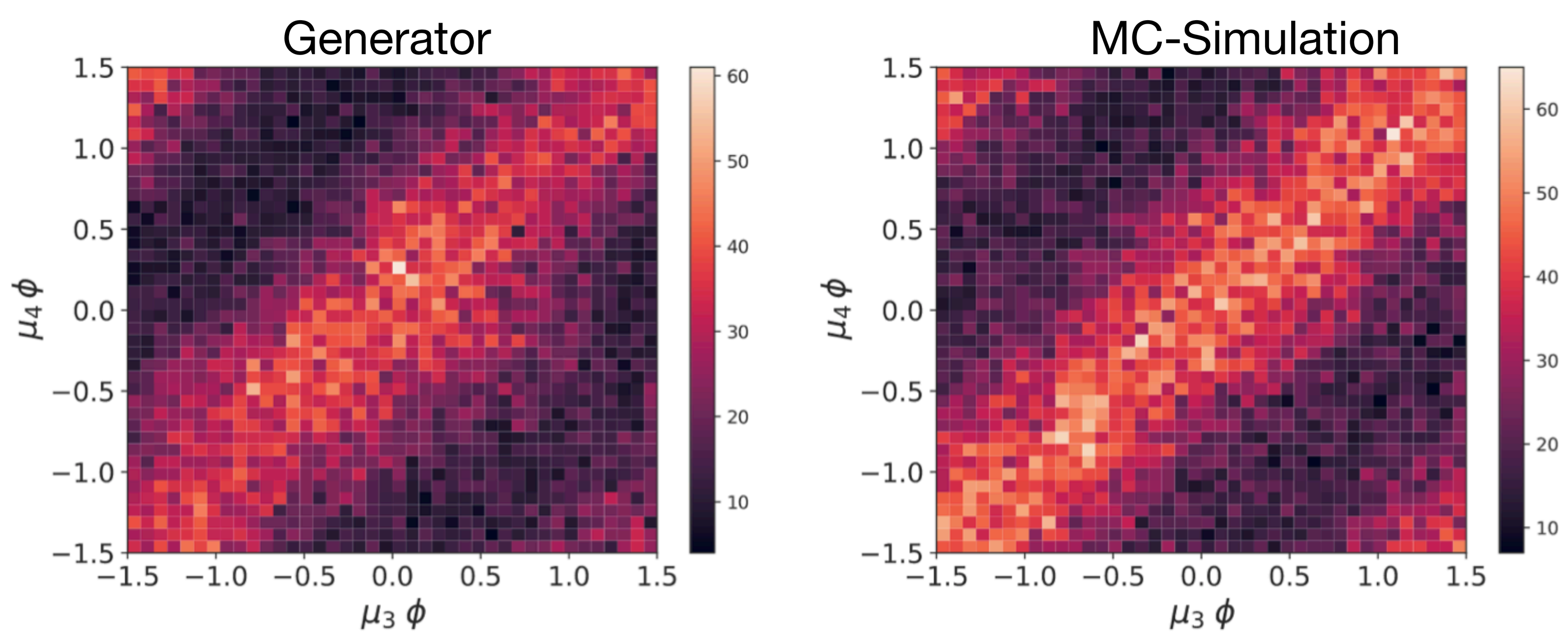
$$k_{\text{Gauss}}(x,y) = \exp -\frac{(x-y)^2}{2\sigma^2} \quad \text{or} \quad k_{\text{BW}}(x,y) = \frac{\sigma^2}{(x-y)^2+\sigma^2}$$

Results

$H \rightarrow ZZ \rightarrow 4\mu$
Madgraph/Pythia + Delphes Simulation

- Final states
  - Basic kinematics are modelled with high precision
  - Angular correlations are similar





- Resonances
- Mass-peaks modelled with minor deviations
- Basic kinematics fit as good